Cyber Safety and Security for Reduced Crew Operations (RCO)

Kevin R. Driscoll
Honeywell International, Inc.
1985 Douglas Drive North
Golden Valley, MN 55422
763-954-6789
Kevin.Driscoll@Honeywell.com

Aloke Roy
Honeywell International, Inc.
7000 Columbia Gateway Dr
Columbia, MD 21046
410-964-7336
Aloke.roy@Honeywell.com

Denise S. Ponchak
Alan N. Downey
NASA Glenn Research Ctr
Cleveland, OH, USA
Denise.S.Ponchak@nasa.gov
alan.n.downey@nasa.gov

***Abstract*—NASA and the Aviation Industry is looking into “reduced crew operations” (RCO) that would cut today's required two-person flight crews down to a single pilot with support from ground-based crews. Shared responsibility across air and ground personnel will require highly reliable and secure data communication and supporting automation, which will be safety-critical for passenger and cargo aircraft. This paper looks at the different types and degrees of authority delegation given from the air to the ground and the ramifications of each, including the safety and security hazards introduced, the mitigation mechanisms for these hazards, and other demands on an RCO system architecture which would be highly invasive into (almost) all safety-critical avionics. The adjacent fields of unmanned aerial systems and autonomous ground vehicles are viewed to find problems that RCO may face and related aviation accident scenarios are described. The paper explores possible data communication architectures to meet stringent performance and information security (INFOSEC) requirements of RCO. Subsequently, potential challenges for RCO data communication authentication, encryption and non-repudiation are identified. The approach includes a comprehensive safety-hazard analysis of the RCO system to determine top level INFOSEC requirements for RCO and proposes an option for effective RCO implementation. This paper concludes with questioning the economic viability of RCO in light of the expense of overcoming the operational safety and security hazards it would introduce.**

TABLE OF CONTENTS

1. INTRODUCTION 1
2. LEVELS OF AUTHORITY DELEGATION 2
3. RCO AUTHORITY QUESTIONS 3
4. RCO INTERFACE TO ONBOARD SYSTEMS 4
5. RCO AIRBORNE SYSTEM ARCHITECTURE 6
6. RELATED RESEARCH AND DEVELOPMENT 7
7. CONTROL HAND-BACK PROBLEMS 7
8. ARE COMMUNICATION THREATS REAL? 8
9. COMMUNICATION ENCRYPTION 9
10. SUMMARY 12
REFERENCES 13
BIOGRAPHY 14

1. INTRODUCTION

During the latter half of the past century, advances in avionics and related technologies have: (1) reduced the total workload of aircraft flight crews and (2) allowed for the reduction of aircraft crew from five-person flight crews in the early 1950s to two-person flight crews in the 1990s. Today, there are research efforts underway for “Reduced Crew Operations” (RCO) or “Single-Pilot Operations”. These phrases, when talking about research, usually relate to FAR (Federal Aviation Regulations) Part 121 operations or equivalents. Some FAR Part 135 operations already are approved for single-pilot operations. The “reduced crew operations” phrase can be read in one of two ways (1) reduced crew-operations or (2) reduced-crew operations, relating respectively to the reduction trends of the past half-century.

The Advanced Cockpit for Reduction of Stress and Workload (ACROSS) study, which is funded, in part, by the European Commission under its Seventh Framework Program, is typical of efforts in this area [1]. It includes objectives covering both interpretations of the “reduced crew operations” phrase. Similarly, this report summarizing Honeywell’s research into cyber safety and security for RCO covers both interpretations of the “reduced crew operations” phrase. However, given the increased safety and security issues of the latter interpretation, this research focused more on the latter interpretation.

An apparently logical extension to the reduced-crew operations trend would be to reduce today’s two-person flight crew down to a single person crew. Significant safety and security hazards will be introduced in the system for RCO when the traditional, two-pilot cockpit is transformed into one pilot operation with support from another person on ground. Shared responsibility across on-air and on-ground personnel will require a highly reliable data communication system that offers very low latency and jitter, as well as high data integrity. In addition, effective protection of the end-to-end information system will be critical to ensure the safety of passengers for passenger aircraft and the survival of the aircraft, crew, and cargo for cargo aircraft.

2. Levels of Authority Delegation

The types and degree of safety and security hazards introduced by an RCO system will depend heavily on the degree of authority that an airborne system relinquishes to the ground and any of its supporting automation. The following subsections describe different degrees of this authority delegation and the ramifications of each, including the hazards introduced, the mitigation mechanisms for these hazards, and other demands on an RCO system architecture.

One thing to keep in mind for these various levels of authority delegation is: what does cockpit resource management (CRM) mean when some of the cockpit resources aren't in the cockpit or anywhere near the cockpit? Much of the existing RCO research has been aimed at questions like this that deal with the human part of potential RCO systems. This report won't revisit this previous research. It will focus mainly on the safety and security aspects of hardware and software; it will cover only the human parts of RCO that haven't been well explored in other research and are tightly tied with the hardware and software. Something that falls into the latter category and is another thing to keep in mind for these various levels of authority delegation is handover effects during changes of authority delegation and whether the airborne crew (AC) or the ground crew (GC) is the pilot-in-command. What exactly happens when more authority is shifted toward the AC or toward the GC? Is there a time period during the handover when neither have control of the aircraft? A more detailed look at this question and related problems for the case of authority being unexpectedly returned to the AC is given in the "7. Control Hand-Back Problems" section below.

In the following subsections, it should be understood that the terms "ground crew" and "GC" refer not only to the totality of the ground component of an RCO system, but also to any airborne automation that supports the ground crew interface.

A possible variation for each of those authority delegation degrees in which the GC is in command includes the addition of an untrained or lesser-trained person in the cockpit who just carries out commands from the GC. This does not represent any difference in the level of authority allocation. Such a variation is best viewed as a "biology-based actuator" for the GC.

2.1. AC is pilot-in-command, GC is just standby redundancy

This is the minimal level of authority delegation. The GC actually has no immediate authority over the aircraft, but the GC has the capability of having its authority elevated to one of the following levels of delegation. An issue here is how that elevation is performed. For RCO operations with a single AC member, this elevation must be able to be performed after the single AC member has become incapacitated. See the "Should the GC be able to take over for an incapacitated AC?" section below for a more detailed discussion of incapacitated crew considerations. It is clear that some form of automation would have to be used to detect that the AC has become incapacitated and to elevate the GC authority to take over from the loss of AC capability. This automated ability to detect incapacitated crew and effect the elevation of GC authority must be a full-time capability. Note that it is incorrect to assume that the dependability requirements for an RCO system can be reduced based on an argument that it is called into play only after the AC has been incapacitated. Even at this lowest level of authority delegation, an RCO system must have the full-time capability of assuming authority over the aircraft. Thus, while an RCO system can be argued to have lower availability requirements due to this argument, the integrity (commission failures) requirements for an RCO system are just as stringent as for any other full-time safety-critical aircraft system. More detailed descriptions of these dependability requirements are given later in this report.

2.2. AC is pilot-in-command, GC is active second pilot

At this level of authority delegation, the GC is another "pair of eyes", sharing the "see and avoid" responsibility with the AC. In addition, the AC may delegate some specific sub-duties to the GC. Again, this calls into question the meaning of RCO CRM. What are the GC's "eyes"? Adding multiple video cameras would require high-bandwidth and potentially safety-critical communication from the aircraft to the ground. There is a question of whether there will be sufficient available bandwidth to support this video traffic. Papers have been written showing that L band communication has insufficient capacity and that C-band would be required. Given that many of the arguments for RCO envision its greatest use in transoceanic flight, this would mean C-band satellites. However, there are no C band satellites. And, there are no plans to create any C-band satellites.

2.3. GC is pilot-in-command, AC is active second pilot

At this level of authority delegation, the GC has immediate full authority over the control of aircraft. In addition to the communication requirements of the previous section, the communication for this level of delegation has an additional requirement for low round-trip latency and jitter (the variation in latency). This communication is also now fully safety-critical.

This level of authority delegation begins to raise the AC recovery time issue. This issue is the amount of time it takes an AC to resume control of the aircraft if GC communications or onboard systems fail. During this time, neither crew is in control aircraft. Again, a much more detailed discussion of this issue can be found in the "Control Hand-Back Problems" section below.

2.4. GC is pilot-in-command, AC is just standby

At this level of authority delegation, the AC is further "out of the loop". The AC would require more recovery time if GC communications or onboard systems fail. There are varying degrees of the AC being "out of the loop". These include being in the cockpit: eating, doing logbook, working on schedule, napping, etc. or being out of the cockpit: lavatory, sleeping, checking on abnormalities, etc. Example scenarios for many of these situations are given in the "Control Hand-Back Problems" section below.

2.5 GC is pilot-in-command, AC is incapacitated/unavailable

At this level of authority delegation, the GC has full authority over the aircraft, with the AC being incapacitated or otherwise unavailable to share in any cockpit duties. Sometimes overlooked when considering AC incapacitation is the fact that the AC's incapacitation may be of a type (for example, seizure or dementia) that would cause them to perform some action(s) that are indistinguishable, at least in part, from the suicidal case described in the next subsection.

2.6 GC is pilot-in-command, AC is an adversary or is suicidal

At this level of authority delegation, the GC has full authority over the aircraft and has to deal with an AC that may be an adversary, such as a hijacker, or an authorized AC member that has become suicidal. After the suicidal hijackings of 9/11, the German wings suicide, and the potential Malaysian MH370 suicide, there have been calls to prevent these kinds of aircraft loss by creating some mechanism for the control of aircraft from the ground. However, there are huge (probably insurmountable) problems with trying to do this. One of these is the fact that the system would have to prevent all the possible ways that an adversarial or suicidal AC could prevent an aircraft from safely completing its flight; and, there a lot of ways that this could be done. Details of this are given in the section below called "RCO Interface to Onboard Safety-Critical Systems". Another problem is that any solution to this scenario creates a new, and probably more dangerous, scenario described in the next subsection.

2.7 AC is pilot-in-command, GC is an adversary

At this level of authority delegation, one can argue that the GC has been given too much authority. If a GC can wrest complete control of an aircraft away from an AC, such a capability could be subverted by someone (inside or outside the system) or a component failure. This could lead to an AC that wants to safely continue the aircraft's flight, but would not be able to do so.

Such a design would violate the "do thy patient no harm" principle by creating a new cyber-attack pathway into the aircraft and another source of natural failures that could adversely affect all safety-critical systems on an aircraft!

3. RCO AUTHORITY QUESTIONS

Should the GC to be able to override a "rogue" AC?

We can group adversarial crew, suicidal crew, and crew that have incapacitation indistinguishable from the latter together into a set called "rogue pilots". An important design decision for an RCO system is to determine if that system should have the ability for the GC to override an AC in case the latter becomes a rogue pilot.

Providing the ability for the GC to override the AC leads to a troubling question: "Who has the ultimate authority, the AC or the GC?" The answer to this must be the same for all situations. Otherwise, who has the authority to decide what the situation is? Whoever/whatever has that decision authority is the ultimate authority. The decision as to who/what has the ultimate authority must be made at the RCO system design time and is fixed for the life of the design. This means that any RCO system design that has a solution for the "GC is pilot-in-command, AC is an adversary or is suicidal" scenario is mutually exclusive to any RCO system design that has a solution for the "AC is pilot-in-command, GC is an adversary" scenario. There are no exceptions to this mutual exclusion. One cannot design an RCO system that can handle both of these scenarios; it has to be one or the other.

Obviously, the ultimate authority would have to be the GC if we want RCO to have the capability for the GC to override the AC. But, why should a GC be any less prone to being rogue than an AC? One can argue that there is a greater probability for a GC going rogue. They don't have to face certain death and they can crash more than one aircraft.

One could envision a redundant GC. But, each member of redundant GC set would have to be totally independent from all other members of that set, including independent communication channels to the aircraft. For this level of GC authority to be viable, the RCO system would have to prevent/mitigate:

- All the possible ways that a rogue AC could make the flight unsafe
- All the failure modes described in section 4
- All the security intrusions that could have a severe safety impact

Should the GC be able to take over for an incapacitated AC?

It is highly likely that this will be required for single-person AC. It is not uncommon to have an incapacitated crew member. In the UK, there are 30 to 40 such incapacitations in a typical year (e.g., 32 in 2009 and 36 in 2004). That is about one for every 10 days. For the number of pilots in this pool, this gives an incapacitation probability of approximately 0.25% per year. This probability is better than the typical requirement for passing a flight physical, which is that a pilot's health should be such that there is no more than a 1% probability that the pilot will suffer an incapacitating health event within a year. One can make a reasonable assumption that other airspaces have similar incapacitation probabilities among its pilots and the number of incapacitation events would be proportional to the number of pilots. Of course, in determining the probability of the incapacitation of an AC in an RCO equipped aircraft, one would have to take into account the fraction of the aircraft in an airspace that are RCO equipped (and for design decision purposes, the number of aircraft that potentially would be RCO equipped).

There are many ways that an incapacitation can cause, and has caused, an AC to inadvertently activate some control that is adverse to safety. These are more than just the Hollywood cliché of an AC having a heart attack and falling onto the stick. There've been a number of instances of seizures, which have caused limb extension; for example, doing a hard-over push on the rudder pedal and then having the foot slip off the rudder pedal and jam it in the hard-over position. There have also been instances of dementia where an aircrew member was unaware of what they were doing.

Given these types of "active incapacitation", solving the AC incapacitation problem is not significantly easier than the malicious rogue AC problem. In fact, we can just duplicate the following bullet list from the above subsection:

For this level of GC authority to be viable, the RCO system would have to prevent/mitigate:

- all the possible ways that a rogue AC could make the flight unsafe
- all the failure modes described in section 4
- all the security intrusions that could have a severe safety impact

Thus, we are left with the following implication chain for the design of an RCO system: single person AC → tolerate incapacitation → assume some adversarial AC action(s)

4. RCO Interface to Onboard Systems

Murphy and Satan

The points at which an RCO system integrates with traditional aircraft systems and the control paths that this integration needs to intercept will depend on the types and degree RCO authority. A failure of a component within an RCO system or a successful external attack into the RCO system can be coupled into traditional safety-critical aircraft systems via this RCO integration. These two sources of RCO-introduced safety hazards can be characterized as "Murphy and Satan" (random naturally occurring failures and failures induced by humans with malicious intent, respectively). Protections must be provided for both and these protections must provide dependability commensurate with the highest criticality level aircraft function that it could adversely affect.

When dependability requirements restrict the probability of failure to be less than 10^{-7} for a one hour exposure (approximately the failure probability of a single integrated circuit), Murphy is indistinguishable from Satan. That is, the worst possible human adversary attack also could be produced by Murphy with help from Mother Nature, with one major exception. This exception is that we assume independent components of a system will fail independently from natural sources, but humans can mount coordinated attacks against multiple components. However, the RCO system interface into existing aircraft safety-critical systems is an exception to this exception. That is, a failure of this interface can appear as a coordinated attack against multiple aircraft safety-critical systems, which had been independent until coupled through the RCO system interface! Thus, the RCO system interface not only would have to be Level A if it is controlling Level A functions, it would have to be what is euphemistically called "Level A+".

Those who are not well-versed in the way that things can fail usually assume that failures are somewhat benign, often consisting only of omission failures. But, when we get down to the low levels of failure probability allowed for safety-critical aviation functions, failure modes can happen that are unbelievable until we find out that they actually do occur. Examples of these can be found in the Real System Failures area of the NASA DASHlink webpages [2].

The design of the RCO system interface into the rest of the aircraft safety-critical systems must be able to tolerate failures of commission (an integrity issue) as well as failures of omission (an availability issue). The same consideration also must be given to the communication link from the GC to the aircraft. We need to find the proper balance between integrity and availability. The reason for this is that fault-tolerance mechanisms that promote one of these characteristics typically is detrimental to the other. To illustrate this, we can look at a simple dual-redundant communication link. Two versions of this link are shown in figure 4-1 below, one designed for availability and the other designed for integrity.

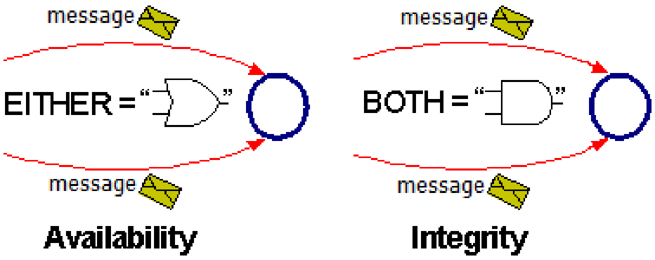

Figure 4-1 Availability versus Integrity

For the highest levels of safety-criticality, in-line integrity mechanism such as checksums and CRCs are insufficient in themselves (see the FAA report DOT/FAA/TC-14/49 "Selection of Cyclic Redundancy Code and Checksum Algorithms to Ensure Critical Data Integrity", the 2005 Dependable Systems and Networks paper "Coverage and the Use of CRCs in Ultra-Dependable Systems", or the web pages at checksumcrc.blogspot.com). For these dual communication link examples, the only fault detection mechanism with sufficient coverage is the comparison of the two messages arriving via the two independent communication paths. What the receiver does with the messages when the messages miscompare depends if the system is designed for availability or integrity. If it needs availability (strive to continue operation), the receiver will arbitrarily select one of the two messages as its input. If it needs integrity (only do correct operation), it will reject both messages. Thus, a dual system designed for availability will accept either message (an OR function) and a system designed for integrity needs to have both messages (an AND function). Note that integrity does not imply safety. For this to be the case, taking no action must be safe. In general, it is not always possible to design a system that has a "failsafe" state and no type of dual-redundant architecture could be designed that would be safe for such systems. A simple dual redundancy can give you availability or integrity, but not both. If both of these characteristics are needed, the system needs to be at least triplex.

This availability vs. integrity observation holds for RCO communications from the ground to the aircraft and its interface into aircraft safety critical systems. First, we need to determine the degree of availability and/or the degree of integrity needed. With sufficient onboard automation, the availability requirements for communication would not be very stringent. The need for this communication is conditional on those events with sufficiently high workload or for AC incapacitation. The probability of this "on-demand availability" working correctly when called upon need not be very high if the probability of it being called upon is low enough. On the other hand, the integrity requirements are not conditional. The ability of the RCO system to contain integrity failures must be full-time. One cannot make an on-demand argument for RCO system integrity, similar to arguments that can be made for systems like autoland. We cannot rely on the AC to turn off the RCO system interface (thus preventing integrity failures) for all times except when they become incapacitated. And, then, when they are incapacitated turn the RCO system on. If an RCO system is designed to detect AC incapacitation, then it must be on all the time.

Even with just these high-level qualitative observations of availability and integrity requirements, we can make some statements about redundancy for RCO communication from the ground.

For availability, simplex (no redundancy) may be sufficient, except for placement of redundant antennas on the aircraft to prevent "shadowing" of the antennas during certain maneuvers where parts of the aircraft may block the RF signal. The final determination of whether simplex is sufficient will depend on the specific demands placed on the RCO system. If communication redundancy is required for availability, each of the redundant communication paths must have sufficient bandwidth to carry the entire RCO communication demand.

On the other hand, integrity requirements would demand at least dual communication redundancy, with the redundant paths possibly being asymmetric. That is, one of the redundant paths would have to carry the entire RCO communication demand, but other paths could just be some compressed version of the entire RCO communication demand and the equivalent of an "enable". This latter capability would be particularly useful when using redundant GC. Note that for asymmetric communication paths, the path(s) with lower demand possibly could be accommodated within existing communication equipment.

Of course, if the system needs redundancy for both availability and integrity, the communication path would have to be triplex. When the shadowing requirement is added to this, we are faced with the extreme demand of finding locations for six antennas on the aircraft.

Traditional Three Layers of Aircraft Control Automation

When looking at suitable locations for where an RCO system would connect into existing safety-critical onboard systems, one likely would begin by looking at the traditional three layers of aircraft control. These can be roughly broken into:

- Flight Management System (planning, source-to-destination profile)
- Auto Pilot (altitude, heading, speed)
- Flight Control (stick and rudder – attitude control, stability) and Engine Control

This list is in "top-down" order, in which each of the upper items in the list provide inputs into the next item lower down the list. Providing RCO inputs into systems only near the top of this list requires less stringent communication latency and jitter requirements than for RCO inputs that are closer to the bottom of this post. However, trying to take advantage of lower communication latency and jitter comes as a trade off with respect to control authority. That is, as the RCO inputs intercept signals in systems toward the lower end of the above list, the greater the authority the RCO system has for taking control away from an AC that may be adversarial or incapacitated. Any design for an RCO system will have to deal with this authority vs. latency trade-off.

Regardless of where the RCO system interfaces into these existing safety-critical systems, the expected dependability (integrity) requirements for the RCO control will be the highest levied on any aircraft system.

Other Potentially Safety-Critical Systems

While the three layers of aircraft control described in previous section are the most often studied and cited locations for an RCO system to interface with other aircraft systems, there are many other controls typically used by an AC that an RCO system may have to control and many of them are safety-critical. Here is a partial list of such controls:

- Power
  - Conversion (AC/DC, DC/AC) and Distribution (tie relays and switches)
  - Circuit breakers (there are a lot of them)
- Fuel distribution (center of gravity control, jettison)
- Flight-control surface trim
- Landing gear
- Spoiler, thrust reverse, and braking systems
- De-icing and pitot heat
- Radio tuning and audio

Each of the items in this list has been implicated as a contributing factor to incidents in which their misuse has led to a catastrophic event. Thus, it should be clear that an RCO system must be able to control all of these. However, these levels of pervasiveness and invasiveness of the RCO interface have not been adequately addressed by previous

RCO and SPO studies (for Part 121) which typically have concentrated on the traditional three levels of aircraft control.

5. RCO AIRBORNE SYSTEM ARCHITECTURE

Depending on the requirements for handling rogue pilots, an RCO system may need to intercept all signals/systems that could possibly cause an aircraft to not continue safe flight (including systems not in the three traditional control layers of FMS, autopilot, and flight control). Even without a rogue pilot (i.e., just "benign" loss of AC) many signals/systems will need to be intercepted to provide a ground override. A couple of possible on-aircraft RCO system architectures are shown in the figures below. Both are expensive, safety-critical, and highly disruptive to many current aircraft systems. These characteristics are unavoidable; they would be true of any RCO airborne system architecture.

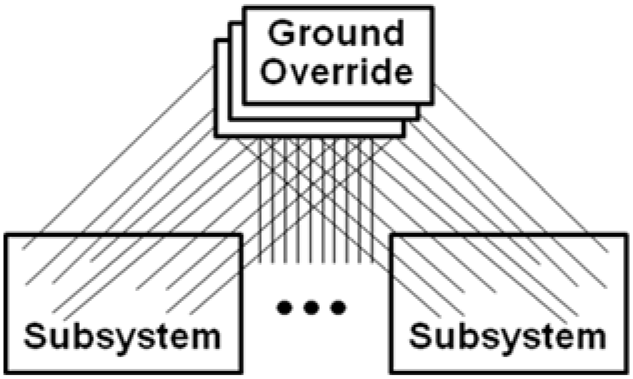

Figure 5-1 Centralized "Porcupine"

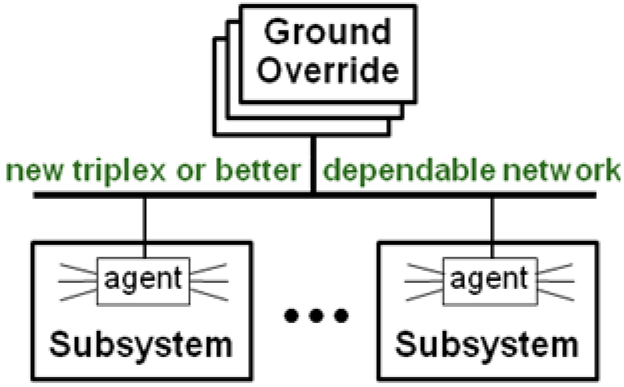

Figure 5-2 Remote Agents

In the "Porcupine" architecture shown in Figure 5-1, individual signals go out from the RCO interface boxes that provide the GC ground override capability to all the points in all the other subsystems where the RCO must intercept some existing signal. The name "Porcupine" comes from the fact that there might not be enough surface area on the Ground Override boxes to accommodate all the possible signal lines.

In the Remote Agents architecture shown in Figure 5-2, the mass of Porcupine wires are replaced by a new high-dependability network that connects the Ground Override boxes to remote agents within each of the other aircraft subsystems. The only difference between this Remote Agents architecture and the Porcupine architecture is the structure of the signal interconnects.

The Ground Override boxes are shown as triplex. This is because many of the places where the other subsystems must be intercepted have no always-safe state. Therefore, a dual-redundant control is insufficient. One example of this is the landing gear. The gear must be up for high-speed flight and down for landing. Neither state (up/down) is safe in the other situation. It should be clear that many of the other safety-critical controls have no universally-safe state.

One set of safety-critical controls might not be so obviously lacking of universally-safe states, but actually is likely to be the set of controls which will make it prohibitively expensive to retrofit an RCO system into an existing aircraft. This set of controls is the circuit breakers. There are a lot of them. If some downstream electrical malfunction could cause a fire, the safe state for the circuit breaker is off. This is the main reason that circuit breakers exist. On the other hand, there are a number of electronic/electrical subsystems on the aircraft for which the safe state is having one or more circuit breakers on. So, dual redundancy isn't enough. And, it's not feasible to make circuit breakers, switches, and relays fail-operational (providing availability and integrity, simultaneously) using triplex. The common way of providing fail-operational capability in circuit breakers, switches, and relays is the quad configuration shown below.

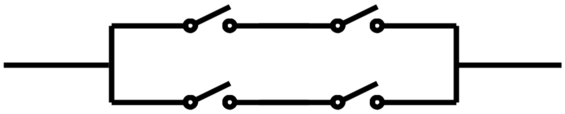

Figure Error! No text of specified style in document.**Quad Circuit Breaker, Switch, or Relay**

This configuration is single fault tolerant to any single stuck-closed or stuck-open component. But, this configuration presents the problem of how to connect a triplex controller to these quad components. The easiest solution is to make the controller quad instead of triplex. Then, each member of the quad controller would independently control one component of the quad circuit breaker. In the Porcupine architecture, the Ground Override boxes would have to be quad and there would have to be four independent control signals from these boxes to each of the quad circuit breakers. In the Remote Agents architecture, the agents would have to be quad and have a triplex-to-quad conversion voting plane between it and the triplex dependable network, or the entire system would have to be quad.

In addition, it is quite likely that many of these intercept points will need to have their actions coordinated. As soon as any type of coordination is required among redundant elements, the possibility of a Byzantine fault is created [3]. To tolerate one Byzantine fault, a minimum of four fault sets is needed [4] Thus, the Ground Override boxes may need to be quad to cover these faults.

6. Related Research and Development

Previous RCO and SPO R&D

While safety, security, and certification issues for the hardware and software that make up the nonhuman parts of an RCO or SPO system haven't been totally neglected in previous studies, these issues certainly have taken a backseat to studying the human parts of these systems. This could be an example of "design procrastination" where the difficult and uninteresting parts of a design problem are delayed to the end of the process. What little publication has been created for these areas of RCO and SPO was consulted for this effort.

To "not reinvent the wheel" and "not look under rocks that have already been examined before", previous R&D in fields adjacent to RCO and SPO were also examined. The two most applicable adjacent fields are unmanned aerial systems (UASs) and autonomous ground vehicles. Both of these areas are currently hotbeds of R&D activity looking at some issues that could be applicable to RCO.

R&D Done In Adjacent Fields

- Unmanned Aircraft Systems (UAS)

While unmanned aerial systems have some issues with availability, safety, and security for remote control of aircraft, there is no AC to share control responsibility and the dependability requirements are much less stringent than for civil transport operations. A few pieces of information from this field are incorporated in subsequent sections of this report.

- (Semi-)Autonomous Ground Vehicles

While ground vehicles don't fly (yet), and they aren't remotely controlled (yet), and their dependability requirements are much less stringent than for aircraft (largely because crashes of ground vehicles don't cause potential drivers and passengers to avoid these vehicles as much as an aircraft crash causes potential passengers to avoid flying), the recent research and development into (semi-)autonomously driven ground vehicles covers an important aspect of RCO system design not present in unmanned aerial systems. This aspect is the large degree of control authority that an in-vehicle person relinquishes to automation and/or remote persons. This degree of control authority delegation is much larger than previous systems (e.g., aircraft autopilots and ground vehicle adaptive cruise control). In fact, we are now looking at situations where the automation/remote control authority supersedes that of the in-vehicle person and that person may not even have the ability to take back some portion of that control.

One of the hot topics in (semi-)autonomously driven ground vehicle R&D is the issue of full autonomy versus shared responsibility. Ford says that the possible interim step to fully autonomous vehicles, where the driving responsibility is shared between an autonomous digital driving system and human drivers, can't be done safely. The problem is the handoff from the digital system back to the human driver when something unexpected happens. Designers can't anticipate every possible situation a vehicle can encounter.

"Right now, there's no good answer, which is why we're
kind of avoiding that space." — Dr. Ken Washington,
Ford's VP of research and advanced engineering

This problem of control being handed back to a human when automation fails is already an emerging problem for cockpit automation systems. Introduction of RCO and/or SPO will exacerbate this problem.

7. Control Hand-Back Problems

NASA's Paul Schutte, in his paper "How to Make the Most of Your Human: Design Considerations for Single Pilot Operations"[12] has the following discourse on "Is Automation the Hero?"

- One reason why computers are so reliable at what they are programmed to do is because they give up at the first sign of trouble.
- When the autopilot reaches its maximum authority, it throws up its hands and tosses control back to the human, whether the human is ready for it or not.
- Pilots routinely must intervene whether it's simply resetting a circuit breaker or turning off the automation.
- The main reason why humans are still on the flight deck is to manage risk by dealing with or avoiding the unexpected, unanticipated, or complex situations

The same things can be said about the RCO communication path from a GC to the aircraft, the path into the aircraft's safety-critical systems, and any of its onboard supporting automation. The issue is that there may be nobody/nothing in control of the aircraft between the time that the communication or automation fails and the time that the AC can retake control of the aircraft. The duration of this time depends on how "out of the loop" AC is at the time of the failure. The following subsections describe these varying degrees of being "out of the loop" and include illustrative events from actual aviation incidents and accidents.

Time needed to get to the controls, when out of the cockpit

On a commuter flight, the captain got stuck in the lavatory due to the door latch being broken. This flight had only one cabin crew member, who had to go into the cockpit when the captain left it. The captain yelled for a passenger to tell the cockpit what was going on. The passenger banged on the cockpit door and yelled through it trying to explain the situation. The problem with this was he had a thick Middle East accent. The people in the cockpit weren't going to open the door under those circumstances. The captain had to breakdown the lavatory door. The flight continued on without further incident after the AC rescinded their radio message that they were potentially in a hijack situation.

A common reason for leaving the cockpit is to investigate an abnormal situation (e.g., smoke). One can argue this is precisely the wrong time to leave the cockpit unattended. The abnormality being investigated could be something that would cause the loss of RCO communication or its interface to critical systems. For example, a half-hour into a scheduled 12-hour flight from SEA to PEK, a cockpit crew member rushed to the rear of the airplane to investigate the smell of smoke, which is never a good sign. On an RCO flight, this would have been the entire crew (!), away from the cockpit for a significant amount of time. The airplane returned to Seattle (see Figure 7-1) for over-night repairs, which replaced a cabin air recirculation fan and one third of the cabin seats.

So, this had to be a nontrivial fire. Such events are not uncommon. Some have said [5] that SPO would require automation that has no hand-back mode (no auto-pilot trip) if the crew has to leave the cockpit.

Figure 7-1 Fire trucks surround an aircraft at SEA

Time needed to get to the controls, when in the cockpit

The first corollary of Murphy's Law is: When things do go wrong, they will go wrong at the most inopportune time. On an Aeroflot Flight 593 (an A310), the captain allowed his two children to sit in the front two cockpit seats. The son accidently disengaged the autopilot lateral control. While there were two members of the cockpit crew in the cockpit, having the children in the way plus the g-forces caused by the lack of the autopilot lateral control prevented crew getting back into their seats and at the controls in time. All 63 passengers and 12 crew members died in the crash.

Once at the controls, time needed to regain situational awareness under normal conditions

In the Air Canada's Incident Report [6] on Air Canada Flight 878 (a B767) had the following to say: "Under the effects of significant sleep inertia (when performance and situational awareness are degraded immediately after waking up)" a pilot mistook the planet Venus as lights of another airplane on a collision course and he dove to avoid it. While this maneuver managed to avoid a collision with the planet, 14 passengers and two crew members were injured because they were not wearing seatbelts.

The cognitive delay due to the "startle effect" is present even when the crew is fully awake.

Audi says its tests show it takes an average of 3 to 7 seconds, and as long as 10, for a driver to snap to attention and take control, even with flashing lights and verbal warnings.

"…anyone who gets behind the wheel [of a semi-autonomous car] must be properly trained. For Audi, this means learning to be a better than average driver…if you need to grab the wheel, the odds are something's gone terribly amiss" [7].

The Air France Flight 447 crash is now well known. It was a scheduled passenger flight from Rio de Janeiro to Paris, which crashed in 2009. The Airbus A330 entered an aerodynamic stall from which it did not recover and crashed into the Atlantic Ocean, killing all 228 persons aboard the aircraft. When the airspeed indicators failed, the autopilot sounded the caution alarm (startle effect) and threw the control immediately to pilots (who were unprepared).

Another crash in which the startle effect was cited as a significant contributing factor was Colgan Air Flight 3407, marketed as Continental Connection under a codeshare agreement with Continental Airlines. It was a scheduled passenger flight from Newark, NJ, to Buffalo, NY, which crashed in 2009. The Bombardier Dash-8 Q400 aircraft entered an aerodynamic stall from which it did not recover and crashed into a house in Clarence Center, NY, killing all 49 passengers and crew on board, as well as one person inside the house.

Recovery time can be even longer if diagnosis is required

The crew of Qantas Flight 32, in which an A380 engine disintegrated [8], needed 50 minutes to sort out all the ECAM warning messages (the crew had no time for ACARS) and assess the aircraft damage. It was lucky that this flight had five cockpit crew members (three normal crew plus a Check Captain and a Supervising Check Captain). So, they had the luxury of having an extra person they could send aft to look out the windows and assess damage. Dealing with abnormal situations may require additional AC, versus a reduction in crew. Richard Woodward (a Qantas A380 pilot and deputy president of the Australian and International Pilots Association) said that the "number of failures is unprecedented […] There is probably a one in 100 million chance to have all that go wrong" [9]. But, there have been over a half-dozen previous similar incidents. The Sioux City DC-10 crash is well known. Again, they were lucky to have additional crew on board, which prevented the crash from being worse than it was.

8. ARE COMMUNICATION THREATS REAL?

When a capability is created to remotely control an aircraft, the security of the communication used for this control is an obvious concern. But, would someone really try to interfere with the flight of an RCO aircraft or is this just a "Hollywood" fantasy? It turns out that the answer is: "Just because you're paranoid, that doesn't mean that they are not out to get you." We have to assume there will be bad actors that are out to get us because there have been a number of instances in the past where radio communications to aircraft have been attacked. Some of these instances are described below in subsections grouped by the type of perpetrators.

Individuals

Officially called a "phantom controller" (a.k.a., "bogus", "fake", or "phony" controller), there are individuals who like to pretend that they are air-traffic controllers. In the UK, there were 18 in 1999. The U.S. will only say that it happens "several times a year". It has been said that these instances have been underreported in order to prevent copycats. This is hard to verify, but from reasonable sources. Jim Epik has written a book on the subject, called "Phantom Controller". He also has created a petition to encrypt ATC communications.

Ad Hoc / Transitory Groups

During the 1981 PATCO strike, some of the striking members became phantom controllers.

Opposing factions in civil wars would like to wrest control of the airspace over their contested country from others involved in the war. And thus, they will interfere with the other factions' communication with aircraft.

Nation-State Sponsored

An Air France captain said that his aircraft received bogus air-traffic control instructions during a flight back to France from Japan. He believed that his aircraft was targeted because he had transmitted a PAN PAN message indicating that an electrical problem had caused half of his cockpit avionics to be inoperative and the crew would be under a heavy workload. The attacker (indications were that it was North Korea) made six attempts to cause the aircraft to fly into an unsafe situation. The captain suggested that encrypting the PAN PAN message for secrecy may have prevented this attack.

9. COMMUNICATION ENCRYPTION

The only really viable current method to protect aircraft communications is the use of encryption. However, there are a number of problems to overcome when employing encryption to protect RCO communications. These problems include each nation's laws governing cryptography, the latency introduced by encryption, and other ways that current encryption algorithms are ill suited for use in real-time cyber-physical systems.

(Inter)National Cryptography Laws

There are laws in almost every country that place some form of restriction on the export, the import, and/or the domestic use of encryption technology. These laws may prohibit the use, limit the use, and/or require licensing for the use of encryption within its territories.

Underlying the use of encryption is a cryptographic key management infrastructure. There are two aspects to key management, trust and logistics.

Trust involves three questions:

1. Who do you trust?
2. With what?
3. To do what?

For example:

1. Can an airline trust the U.S. government?
2. With its cryptographic keys?
3. Not to reveal these keys (to North Korea, UK, Israel?)

Note that specifics are important. In the last element of the example, some airlines might view the U.S. revealing cryptographic keys to Israel as trustworthy; where as other airlines would consider that to be untrustworthy. Most airlines would expect that the U.S. would not reveal its cryptographic keys to North Korea.

Key management logistics are the mechanisms to enforce the trust. This includes the creation of keys with their ownership association, key distribution, and revocations. Key management allows only authorized users to have possession of private or secret keys, often only for a set period of time. These cryptographic keys can have an ordinary use (e.g., RCO communications) and an extraordinary use (e.g., a government investigation).

The laws governing cryptography in many countries require that some arm of the country's government have access to the "plaintext" that has been encrypted. The cryptographic literature uses the term "plaintext" to mean anything that will be encrypted or has been decrypted and the term "ciphertext" is used to mean the equivalent of the plaintext after it has been encrypted but before it has been decrypted.

Usually, the easiest way to give a government access to the plaintext is to allow them access to the cryptographic keys used for the encryption. This will allow the government to decrypt the cipher-text. But, this still can complicate the key management infrastructure.

Much of the literature covering future encryption systems for aircraft communications assumes that just saying an X.509-based public key infrastructure (PKI) will be used for key management is a sufficient explanation for how the key management problems will be solved. But, full PKI is heavy weight and doesn't solve all the problems by itself.

PKI doesn't answer the trust questions. The trust questions include the question of whose keys will be used for a particular flight. As a complex example to illustrate the point, let's say that an aircraft manufacturer includes some cryptographic equipment manufactured by some avionics supplier; the aircraft manufacturer sells the aircraft to a leasing company; the leasing company leases the aircraft to a scheduled airline; the airline rents the aircraft to a charter company at times when the airline isn't using that aircraft; and, the charter company hires a crew that normally works for some other rival airline. Whose keys should be used for the encryption? Should the cryptographic equipment have software with a dedicated link to some key management infrastructure owned by the avionics company? the aircraft manufacturer? the leasing company? the airline? Or, should the crew load keys as part of preflight? If so, what keys should be used? the charter airline's keys? the crew's personal keys? the keys they use as employees of the rival airline? Should there be one set of keys for all systems on the aircraft that want to communicate with the ground? Or, should some systems have keys (or use a key infrastructure) that is different from other systems? For example, the manufacturer of engines that are leased in a "power by the hour" arrangement might like to have engine performance data transmitted to them using their own key. Should the keys used for RCO communication be the same as used for CPDLC?

Encryption can be used to provide secrecy and/or authentication. These two properties don't need to be tied together. Often glossed over in discussions of key management is the fact that key distribution needs secrecy protection for private and secret keys, even if these keys are only used for authentication (not secrecy). Popular authentication schemes need private keys (for public-key system signatures) or secret keys (for message authentication codes). The need for secrecy in the distribution of these keys complicates the key management infrastructure and can cause problems with national laws that restrict encryption used for secrecy, when an encrypted channel is used to provide secrecy for key distribution rather than using physically secure communication path for the key distribution.

There are current uses of PKI for aircraft communication. However, it is unknown whether this PKI can be used for RCO communication.

During the course of this study, an invention was created to mitigate some of the issues for key management logistics and potential legal problems for aircraft communication encryption.

Encryption Latency

One problem encountered by UAS operations is communication latency. The sum of the communication latencies can be on the order of a couple seconds, which can make closed-loop remote control of an aircraft difficult. Encryption of this communication can be an aggregating actor in these latencies. The communication for each iteration around the closed loop incurs the latency of two encrypts and two decrypts (the four arrows in Figure 9-1).

If AES (or similar block cipher) is used to provide secrecy and integrity, a block (e.g., 128 bits) of store-and-forward latency has to be added, plus the latency for any added initialization vector (IV) and/or integrity data (e.g., 32 bits each). These latencies depend on communication speed (the slower the link, the longer these latencies) and they have to be added to the cryptographic computation latencies. The sum of these latencies doubles if handshakes (e.g. ACK/NAK) are used and are encrypted.

UASs try to mitigate this cryptographic latency problem by using very high-speed (e.g., 10 Gbps) communication links and special hardware encryption (e.g., KG-340 encryptors and Single-Chip Crypto field programmable gate arrays). It is unlikely that RCO communications can find such a wide bandwidth for its use and adding additional high-speed encryption hardware can be expensive.

Problems with general-purpose cryptography in cyber-physical systems

An RCO system is an example of a cyber-physical system with real-time and other constraints not seen in general-purpose processing. While latency and jitter may be the main differences in requirements/constraints between general-purpose processing and cyber-physical system processing, there are a number of other problems with employing general-purpose cryptographic algorithms in cyber-physical systems. Many of these problems compound the latency and jitter problems.

The remainder of this subsection deals with symmetric encryption algorithms implemented in software, possibly with hardware support in the form of instructions in the processor's instruction set architecture (ISA) or an adjunct crypto field programmable gate array. The temporal performance of asymmetric (public key) algorithms is not critical for RCO communications. This is because the use asymmetric cryptography can be restricted to the exchange of keys that will be used in symmetric encryption algorithms to achieve secrecy, continuing authentication, and/or integrity (via message authentication codes); and, the key exchange can be perform during pre-flight or at other times when temporal performance is not important. Also not discussed are stand-alone encryption "boxes", because their added costs in terms of cash outlay, size, weight, power, and latency makes them less desirable.

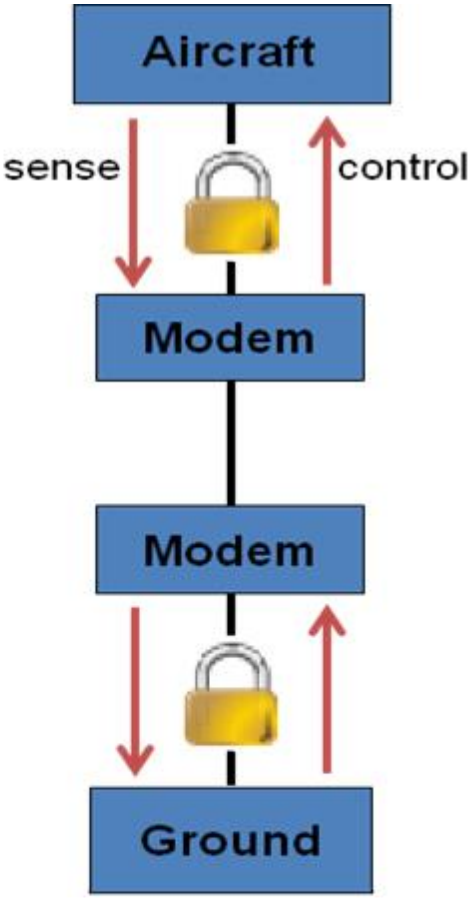

Figure 9-1 Encryption Latency

One of the problems with employing general-purpose cryptographic algorithms in cyber-physical systems is the slow startup for each key change. The startup delay is due to "key scheduling" being done. Key scheduling is the conversion of the cryptographic key into data that the encryption algorithm uses internally. When the encryption is not used for real-time cyber-physical systems, it makes sense for key scheduling to be made expensive. The rationale is that legitimate users incur these startup delay costs a relatively small number of times. On the other hand, an attacker that uses some form of brute force key-related search would have to try a huge number of different keys and incur the startup delay cost a vastly greater number of times. This is one of the reasons why general-purpose cryptographic algorithms have been designed with key scheduling that is slow. Another reason that general-purpose cryptographic algorithm key scheduling is slow is that they have been optimized for peak throughput performance, which is usually measured in clock-cycles-per-byte. In the "my algorithm is faster than your algorithm" speed propaganda wars, startup doesn't count. Therefore, to game the system, algorithm designers can put more work into the startup to make the rest of the algorithm run faster.

As another point of comparison, communications in cyber-physical systems typically use smaller messages and sessions than communications in general-purpose systems. This means that cyber-physical system communications have less data over which to amortize startup costs.

With the cryptography speed propaganda focused on peak throughput, average throughput and worst-case throughput are ignored. However, in cyber-physical systems, typically only the worst case timing counts, average is unimportant and peak is even less important. A missed real-time deadline cannot be helped by finishing early at other times. For cyber-physical systems, latency and jitter are both usually more important than throughput. Often, jitter is more important than latency because control algorithms can better deal with a known latency rather than with instances of unknown jitter.

All known general-purpose crypto algorithms need to store some data in memory while they're executing. Cyber-physical software generally makes heavy use of multitasking with many context switches per second that can cause each task's cache entries to be evicted (replaced with some other task's data). To guarantee timing performance, one must assume that most memory accesses will cause cache misses. But, existing crypto performance propaganda assumes a "pre-warmed" cache. That is, timing performance measurements are done only after making sure that all the data the algorithm possibly could use are in the cache. Not only does the dependence on cached data adversely affect temporal performance, it also can be (as has been) a path that leaks information that can be used to "break" the encryption. Even if all needed data are in the L1 cache, cache *hits* can be expensive (equivalent to about a half-dozen instructions in an Intel i7 processor). The only solution to these cache problems is to not use cache, which means not using any data during the execution of an algorithm that does reside totally with the register set of the processor.

A cryptographic algorithm characteristic closely related to low-latency is "key agility". This is the ability of an algorithm to easily and quickly change from one key to another. There are two types of key agility: the first is the ability to switch to a new key and/or a key with a new IV that hasn't undergone any key scheduling, the second is the ability to switch to a key and IV pair that has undergone any required key scheduling. Of course, the former requires that key scheduling be done and algorithms needing good key agility must minimize the time and effort required to do key scheduling. The second type of key agility depends on the amount of data that the algorithm must use during its execution, therefore, algorithms needing good key agility should minimize the amount of data it needs while executing. For avionics in general, good key agility may be required if different subsystems and/or applications within the aircraft need different keys and there is a centralized provider of encryption services for these subsystems and/or applications. Good key agility also may be required as an aircraft crosses boundaries that delineate jurisdictions where different keys must be used and the keys must be quickly changed at the boundary in order to avoid a communication "dead zone" where encrypted communication can't be performed.

General-purpose crypto algorithms increase the sizes of the messages they encrypt. This increase can include data needed as an IV, padding, and integrity check data. This need is amplified by the small sizes of many cyber-physical system messages, where even small per-message overhead due to cryptography can be a large burden. This may require more communication bandwidth than is available. A design goal for a real-time cyber-physical cryptographic algorithm is to minimize or eliminate message expansion. This requirement eliminates the use of those block cipher modes that round messages up to the next block size.

In order to provide the properties of continuing authentication, secrecy, and integrity, most existing cryptographic systems use separate secrecy and integrity algorithms or use an added integrity mode that is wrapped around a secrecy algorithm. Compared to an authenticating encryption (AE) (a.k.a., integrity-aware) algorithm, which intrinsically provides continuing authentication and integrity along with secrecy, these approaches exacerbate the problems discussed here.

These are the reasons, we created an AE algorithm (called BeepBeep) specifically for real-time cyber-physical and/or retro-fit applications. This algorithm was created in 1999 and published in 2002 [10]. It has a low code size, zero working data memory, low latency, good key agility, and provides continuing authentication, secrecy, and integrity in a single pass. In the last couple of decades, there have been several competitions and initiatives to create new encryption algorithms, e.g. AES [11], CRYPTREC [12], eSTREAM [13], and NESSIE [14]. But, none of them had explicitly stated goals that addressed the problems discussed here.

10. SUMMARY

The RCO concept does not seem to be economically viable for Part 121 operations, at least in the short term where existing aircraft would have to be retrofitted for RCO capability. This follows from the observation that: single person AC → tolerate incapacitation → assume some adversarial AC action(s) … and, the very high cost of implementing an RCO system that can safely and securely provide the capability of controlling an aircraft in which some actions by an incapacitated crew could be similar to that of an adversary.

The cost calculus assumes that the AC to be replaced will be a First Officer, costing a typical salary of just over $100,000 a year plus benefits. However, 100% of that cost cannot be eliminated. There must be some GC costs. If there is a 1:1 replacement of AC with GC, obviously, there is no labor cost savings. There have been estimates that a GC can handle five aircraft simultaneously. But, that must be for benign conditions and the GC intercepting the aircraft systems at the FMS level (possibly, the Autopilot level). If the GC has to intercept the aircraft systems at the Flight Control level (required for adversarial or incapacitated AC), it is hard to imagine that the aircraft:GC ratio can be better than 1:1. This suggests that an RCO system installation on an aircraft must have interception points at the lowest level (for full control) and at a higher level (to ease latency constraints, reduce the number of crew actions that need to be taken, and allow a greater aircraft:GC ratio). A potential GC structure would be to have the number of GC be: number-of-aircraft-to-be-controlled / 5 + 1. This assumes that no more than one aircraft within the set of aircraft to be controlled would need continuous control at the Flight Control level at any point in time. If a GC center is designed to control 20 aircraft simultaneously, the GC complement would have to be 20/5 + 1 = 5. To this number, we need to add another GC to allow for breaks. This ignores any additional personnel needed to protect against adversarial GC. With 6 GC under RCO replacing 20 AC First Officers under today’s two person ACs, this is a reduction of crew cost of $70,000 plus 70% of benefits per year per aircraft.

The cost of completely redesigning and replacing most of the cockpit avionics and adding a quad-redundant (or better) Ground Override system, that has tentacles into many locations within most of these systems (many of which will also have to be at last quad redundant), will be more than the cost for original avionics and will have fewer aircraft over which to spread the development cost. This development cost also will be higher than the original development cost due to the Ground Override system needing to be, euphemistically, “Level A+” because of its potential to be a single point of failure for all of the critical avionics. In addition to these aircraft costs, we must add the development, deployment, and operation cost for the ground segment. These “ground” costs could be very large when they include the development, deployment, and operation costs of a C-band satellite system. The amortization of all these large costs (including time value of money) must be less than the crew labor cost saved.

It should be noted that some sources believe that RCO/SPO may have some cost benefit. For example, the account of Dr. R. Mike Norman’s presentation at NASA’s Single-Pilot Operations Technical Interchange Meeting [15] includes: “SPO may have economic benefit, but once again, new costs associated with SPO were not addressed”. The latter half of this quote indicates that the magnitude of the cost to provide coverage for the safety and security problems identified in this report were not accounted for in this assessment.

The inclusion of RCO within future aircraft designs would cost less than for retrofit. There are two reasons for this. The first is that some avionics developments will make it easier to add RCO functionality, just as a byproduct of their creation for other reasons. A good example of this is the replacement of individual circuit breakers with an integrated “electronic fuse box”. This will make it a lot easier for an RCO Ground Override interface to control the equivalent of circuit breakers. The second reason is that future avionics can anticipate the possible addition of RCO. However, the degree to which creators of avionics would be willing to add “hooks” for an RCO option is unknown, given that these “hooks” would add some cost for all same-type aircraft, including aircraft that don’t use RCO. It is unclear if the reduced cost for RCO in future aircraft would make RCO economically viable.

References

[1] European Union. http://www.across-fp7.eu [Accessed Sept 30, 2016]

[2] K. Driscoll. http://c3.nasa.gov/dashlink/resources/624 [Accessed Sept 30, 2016]

[3] K. Driscoll *et al*. "The Real Byzantine Generals." *Proc. of the 23rd Digital Avionics Systems Conference (DASC).* 2004, pp. 6.D.4-1 - 6.D.4-11

[4] L. Lamport *et al*. "The Byzantine Generals Problem." *ACM Transactions on Programming Languages and Systems*, Vol.4, No.3, July 1982, pp.382-401.

[5] D. Learmount. https://web.archive.org/web/20100715144540/http://www.flightglobal.com/blogs/learmount/2010/06/the-lonely-airline-pilot.html [Accessed Sept 30 2016]

[6] Air Canada Aviation Investigation Report A11F0012

[7] A. Davis. www.wired.com/2015/01/rode-500-miles-self-driving-car-saw-future-boring [Accessed Sept 30, 2016]

[8] Australian Transport Safety Bureau, Aviation Safety Investigations & Reports, AO-2010-089

[9] K. Schneider. http://www.news.com.au/travel/travel-updates/qantas-jet-could-have-exploded/story-e6frfq80-1225956388231 [Accessed Sept 30, 2016]

[10] K. Driscoll. "BeepBeep: Embedded Real-Time Encryption." *Fast Software Encryption 2002 / Lecture Notes in Computer Science*, 2002, Vol. 2365 pp164-178

[11] https://en.wikipedia.org/wiki/AES_process [Accessed Oct 30, 2016]

[12] http://www.cryptrec.go.jp/english [Accessed Oct 30, 2016]

[13] http://www.ecrypt.eu.org/stream [Accessed October 30, 2016]

[14] https://www.cosic.esat.kuleuven.be/nessie [Accessed Oct 30, 2016]

[15] Section 5.6.2 of NASA/CP—2013–216513 "NASA's Single-Pilot Operations Technical Interchange Meeting: Proceedings and Findings" April 2013.

BIOGRAPHY

***Mr. Kevin R. Driscoll** is a Fellow in Honeywell's Research Labs with over 40 years' experience in safety and security critical systems. He was the chief avionics architect for NASA's Orion CEV and co-architect for the Boeing 777 cockpit. He was a principal designer of the SAE AS4710 PI-bus and the ARINC 659 SAFEbus, the only two backplane bus standards with significant fault tolerance. He helped design the bus that became the IEEE 1149 JTAG test bus. He led the effort to create the FAA "Handbook for Data Network Evaluation Criteria". He also contributed to the digital architecture of the U.S. National Aerospace Plane (NASP), Space Defense Initiative (SDI), Advanced Launch System, and Honeywell's Vetronics programs and unmanned underwater vehicles. Prior to joining Honeywell, he worked in the areas of voice and data cryptography for the U.S. Army Security Agency and has developed cryptography specifically for real-time systems. Mr. Driscoll has 50 patents issued or pending covering safety and security critical real-time systems. He is a member of the IEEE/IFIP WG 10.4 on Dependable Computing and Fault Tolerance. He was the U of MN CSci Distinguished Alumnus for 2011-2012.*

***Mr. Aloke Roy** is a Senior Program Manager with Honeywell Advanced Technology organization. He currently manages data communication, information security and radio technology development programs supporting Honeywell Aerospace. Previously, Mr. Roy was Director of Programs at Flextronics Corporation managing several major telecommunications OEM accounts. In this role, Mr. Roy was responsible for business development, outsourcing, and globalization of hardware design activities supporting large volume contract electronic manufacturing. His prior experiences include various positions at AT&T Bell Laboratories and ARINC Aviation Systems Division. As Systems Engineering Director at ARINC, Mr. Roy oversaw development of SATCOM, HF, VDL, ATIS, and PDC standards and services. Currently, Mr. Roy chairs ICAO ACP Working Group 'S' and RTCA Special Committee 223, which are developing the Aviation Internet Protocol and Aeronautical Mobile Airport Communication System requirements and operational performance standards. Mr. Roy holds several patents on aeronautical, wireless and secure communications.*

***Ms. Denise S. Ponchak** is the Deputy Branch Chief of the Communications Architectures, Networks and Systems Branch at the National Aeronautics and Space Administration's (NASA) Glenn Research Center at Lewis Field in Cleveland, Ohio. The Branch is responsible for designing advanced networking concepts, architectures, technologies and system integration for aeronautics and space applications. Prior to becoming a supervisor, Ms. Ponchak was an Aeronautical Communications Project Manager focusing on increasing the National Airspace System's telecommunications capability, and a communications research engineer supporting future satellite-based communications. She holds a Bachelor's of Electrical Engineering and a Master's of Science in Electrical Engineering from Cleveland State University in 1983 and 1988 respectively. Denise is a Foodie who likes to hike, ski and travel.*